CICLO DE VIDA DE SOFTWARE

ANÁLISE DE SISTEMAS

Qualificar um produto é muito bom para que tenhamos certeza de que há seriedade e preocupação com a satisfação em tê-lo, mas, qualificar o processo de produção é mais importante para obter um produto melhor. Ambas as qualificações (da produção e do produto) são largamente utilizados na produção de muitos produtos inclusive no desenvolvimento de softwares.

Hoje, temos normas que certificam o processo de produção de software bem como o software pronto. Tais normas exigem cada vez mais qualidade no gerenciamento do projeto e tais exigências são convertidas em benefícios para os usuários e desenvolvedores. Todo desenvolvimento de um software é caracterizado por fases, que quando colocadas em sequência, obtêm-se um **Ciclo de Vida do Sistema** e é este ciclo de vida que deve ter qualidade.

QUALIFICAÇÃO DO **PROCESSO DE DESENVOLVIMENTO** DO SOFTWARE

Como pode ser esperado, as menores firmas costumam ser relativamente informais: os projetos são iniciados como resultado de uma discussão verbal entre o usuário e o gerente do projeto, e o projeto prossegue a análise de sistemas até o projeto e implementação sem muita algazarra.

Em grandes organizações, no entanto, as coisas são feitas em uma base muito mais formal.
As várias comunicações entre os usuários gerência e equipe do projeto costumam ser documentados por escrito, e todos entendem que o projeto passará por diversas fases antes que seja terminado. Ainda assim existem grandes diferenças entre o modo com que dois gerentes de projetos na mesma organização conduzem seus respectivos projetos.

Normalmente fica a cargo do gerente de projeto determinar de que fases e atividades o seu projeto consistirá e como essas fases serão conduzidas.

Como o cliente explicou

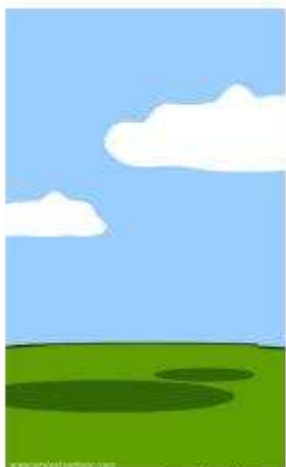
Como o lider de projeto entendeu

Como o analista planejou

Como o programador codificou

O que os beta testers receberam

Como o consultor de negocios descreveu

Valor que o cliente pagou

Como o projeto foi documentado

O que a assistencia tecnica instalou

Como foi suportado

Quando foi entregue

O que o cliente realmente necessitava

Recentemente o método assumido para o desenvolvimento de sistemas começou a mudar. Mais e mais grandes e pequenas organizações estão adotando um único e uniforme ciclo de vida do projeto, ou simplesmente "o modo com que fazemos as coisas por aqui".

Normalmente contido em um livro tão exuberante quanto o manual de padrões que se encontra (fechado) na mesa de cada programador, o ciclo de vida documentado do projeto fornece uma forma comum para que todos na organização passem a entender como pode ser desenvolvido um sistema de computador.

O método pode ser caseiro, ou alternativamente, a organização pode decidir comprar um pacote de gerenciamento de projeto e depois moldá-lo às necessidades da companhia.

As principais razões de se ter um ciclo de vida de projeto são:

- ✓ Definir as atividades a serem executadas em um projeto.
- ✓ Introduzir a coerência entre muitos projetos na mesma organização
- ✓ Fornece pontos de checagem para controle de gerência e pontos de checagem para a decisão "ir / não ir".

Não existe uma regra ou um ciclo de vida padrão para desenvolvimento de sistemas. Você pode usar um ciclo pré-definido por um determinado autor, pode usar o ciclo de um autor e moldá-lo conforme seu trabalho ou ainda criar seu próprio ciclo.

Porém, a escolha deve ser feita analisando os fatores:

- ✓ Características do fornecedor (equipe de desenvolvimento)
- ✓ Habilidade - Analisar os conhecimentos técnicos da equipe.
- ✓ Recursos - Equipamentos e material humano
- ✓ Características do cliente (usuário)
- ✓ Visão - Conhecimento de suas necessidades
- ✓ Tempo - Aceitável para a implantação
- ✓ Recurso financeiro - Ter disponibilidade financeira para o investimento em software e hardware
- ✓ Insegurança - Confiabilidade nos resultados a serem gerados pelo sistema

Modelo cascata

Esta abordagem baseia-se no modelo cascata ou método linear de desenvolvimento

O ciclo é representado pelas seguintes fases:

- ✓ Levantamento: definição preliminar do escopo do sistema, restrições e conceitos alternativos;
- ✓ Análise: especificação funcional do sistema (Projeto Lógico);
- ✓ Projeto: especificação completa da arquitetura de hardware e software, estruturas de controle, estruturas de dados do sistema, interfaces;
- ✓ Codificação: codificação e teste individual dos programas;
- ✓ Teste: teste dos componentes integrados do sistema;
- ✓ Implantação: implantação de maneira gradativa, a fim de evitar insatisfação e possibilitando a correção do sistema. Implantação piloto / paralela e definitiva;
- ✓ Operação e Manutenção: utilização do sistema e modificações decorrentes de erros e mudança de necessidades.

Vantagens deste modelo:

- ✓ O modelo cascata é apropriado para sistemas transacionais onde as rotinas e procedimentos a serem automatizados são altamente estruturados.
- ✓ Só avança para a tarefa seguinte quando o cliente valida e aceita os produtos finais da tarefa atual

Desvantagens deste modelo:

- ✓ É difícil capturar os requisitos de uma só vez;
- ✓ Alto custo de correção das especificações quando nas fases de Teste e Implantação.
- ✓ Nesse ciclo quase não existe oportunidade para o usuário realizar alteração em pontos dos requisitos congelados. As atividades são realizadas em seqüência e não existem retornos entre as atividades. Assim, fica evidente que os projetos realizados com este ciclo de vida se caracterizam pela alta incidência de manutenção, pois estão sujeitos a poucas alterações durante o desenvolvimento.

CUIDADO!

Lavantamento
Análise
"Mar de Rosas"
Alto
Investimento
Projeto
Codificação
Dificuldades
Documentação
Testes
Incerteza
Implantação
Insatisfação
Manutenção
Ciclo de Vida
em Cascata

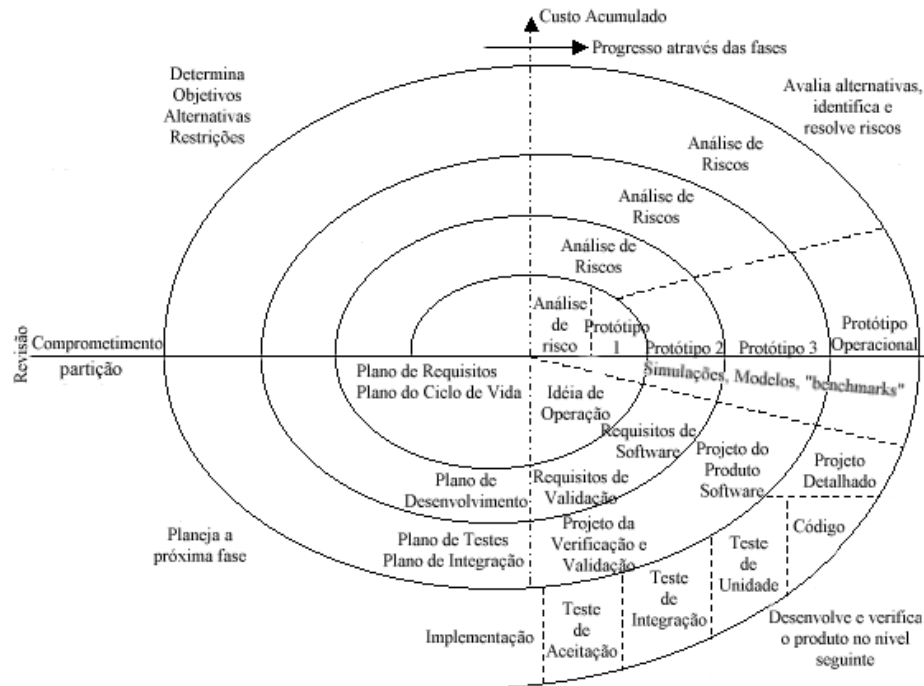
FITO
FUNDAÇÃO INSTITUTO TECNOLÓGICO DE OSASCO

Neste modelo o projeto é atacado como uma série de pequenos ciclos, cada um finalizando uma versão de um software executável.

O modelo em espiral foi proposto por Boehm em 1988 como forma de integrar os diversos modelos existentes à época, eliminando suas dificuldades e explorando seus pontos fortes.

Este modelo foi desenvolvido para abranger as melhores características tanto do ciclo de vida clássico como da prototipação, acrescentando, ao mesmo tempo, um novo elemento - a análise de riscos - que falta a esses paradigmas.

Entretanto a integração não se dá através da simples incorporação de características dos modelos anteriores. O modelo em espiral assume que o processo de desenvolvimento ocorre em ciclos, cada um contendo fases de avaliação e planejamento, onde a opção de abordagem para a próxima fase (ou ciclo) é determinada.

Modelo espiral

O modelo original em espiral organiza o desenvolvimento como um processo iterativo em que vários conjuntos de quatro fases se sucedem até se obter o sistema final.

1. Primeira fase (determinação de objetivos, alternativas e restrições): Ocorre o comprometimento dos envolvidos e o estabelecimento de uma estratégia para alcançar os objetivos.
2. Segunda fase (análise e avaliação de alternativas, identificação e solução de riscos): Executa-se uma análise de risco, por exemplo, com prototipação. Se o risco for considerado inaceitável, o projeto pode ser abortado.
3. Terceira fase (desenvolvimento do produto): Neste quadrante pode-se considerar o modelo cascata.
4. Quarta fase (avaliação do produto): O produto é avaliado e se prepara para iniciar um novo ciclo.

Vantagens deste modelo:

- ✓ Modelo em espiral permite que ao longo de cada iteração se obtenham versões do sistema cada vez mais completas, recorrendo à prototipagem para reduzir os riscos.
- ✓ Este tipo de modelo permite a abordagem do refinamento seguido pelo modelo em cascata, mas que incorpora um enquadramento iterativo que reflete, de uma forma bastante realística, o processo de desenvolvimento.

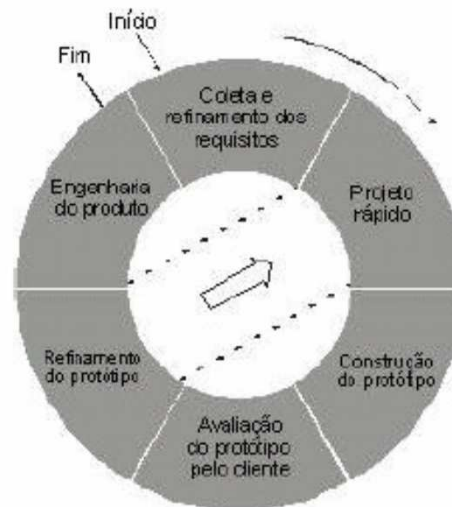
Desvantagens deste modelo:

- ✓ Pode ser difícil convencer grandes clientes (particularmente em situações de contrato) de que a abordagem evolutiva é controlável.
- ✓ A abordagem deste tipo de modelo exige considerável experiência na avaliação dos riscos e baseia-se nessa experiência para o sucesso. Se um grande risco não for descoberto, poderão ocorrer problemas.

O modelo de prototipação se baseia na utilização de um protótipo do sistema real para auxiliar na determinação de requisito.

Um protótipo deve ser de baixo custo e de rápida obtenção, para que possa ser avaliado.

Para isto, uma determinada parte do sistema é desenvolvida com o mínimo de investimento, mas sem perder as características básicas, para ser analisada juntamente com o usuário.

Um protótipo é um exemplo de como será a solução (e não pode ser usado como solução).

Orientações

Não é necessário entregar a atividade ao professor

Escreva a pergunta e resposta em seu caderno

Organize seu material

Toda atividade vale nota. Assim que terminar chame o professor para garantir seu ponto

Questões

1. Para que serve o ciclo de vida do software?
2. O que deve ser considerado na escolha do ciclo de vida do software?
3. Em que consiste o modelo em Cascata? Cite vantagens e desvantagens.
4. Em que consiste o modelo em Espiral? Cite vantagens e desvantagens.
5. Em que consiste o modelo de prototipação?